

Static Malware Analysis

Nouran Alotaibi

02/03/2025

REMnux

Site: remnux.org



[HOME](#)

[DISTRO](#)

[CONTAINERS](#)

[PEOPLE](#)

[DOCS](#)

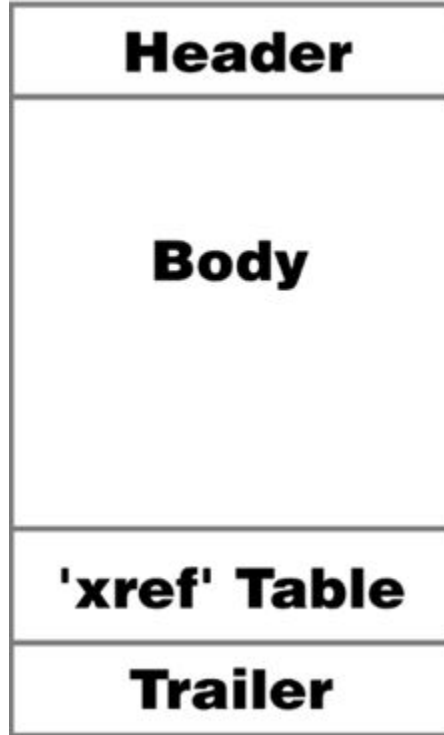
REMnux: A Linux Toolkit for Malware Analysis

REMnux® is a Linux toolkit for reverse-engineering and analyzing malicious software. REMnux provides a curated collection of free tools created by the community. Analysts can use it to investigate malware without having to find, install, and configure the tools.



Analyzing PDF Files

Structure of a PDF file



PDF File Header

Irrespective of the PDF version, a PDF file starts with a header containing unique identifier for PDF and the version of the format such as %PDF-1.x where x ranges from 1-7.

File Body

In the body of the PDF document, there are objects that typically include text streams, images, other multimedia elements, etc. The Body section is used to hold all the document's data being shown to the user.

Cross-Reference Table

The cross-reference table contains information that permits random access to indirect objects within the file so that the entire file need not be read to locate any object.

File Trailer

The trailer of a PDF file enables a conforming reader to quickly find the cross-reference table and certain special objects. Conforming readers should read a PDF file from its end.

Static Malware Analysis of PDF files:

The static analysis within REMnux is usually done in 3 parts.

1. Identifying suspicious objects using **pdfid** tool in Remnux. Objects could be an image, javascript, forms, text contents etc.
2. Identifying the reference number or object ids (the numbers pointing to the objects of interest) using **peepdf** tool
3. Dumping the raw contents from the suspicious objects using **pdf-parser**. Once the contents of the objects are dumped we can then analyze through and use external threat intel tools to investigate further.

Identify Suspicious Objects Using pdfid

```
pdfid.py <filename with full directory>
```

```
remnux@remnux:~/Documents/analysis$ pdfid.py '/home/remnux/Documents/analysis/025ba9ce4a2118a9ca7b115c8869ff73bc16bad3732ba359cef1e60ad8f961f9'
PDFiD 0.2.8 /home/remnux/Documents/analysis/025ba9ce4a2118a9ca7b115c8869ff73bc16bad3732ba359cef1e60ad8f961f9
PDF Header: %PDF-1.4
obj                49
endobj             49
stream            8
endstream         8
xref              2
trailer           2
startxref         2
/Page             2
/Encrypt          0
/ObjStm           0
/JS               0
/JavaScript        0
/AA               0
/OpenAction       0
/AcroForm         0
/JBIG2Decode      0
/RichMedia        0
/Launch           0
/EmbeddedFile     0
/XFA              0
/URI              26
/Colors > 2^24    0
```

Identify the Reference Number or Object ids for the Identified Suspicious Object Using peepdf Tool

Peepdf <filename with full directory>

```
remnux@remnux:~/Documents/analysis$ peepdf '/home/remnux/Documents/analysis/025ba9ce4a2118a9ca7b115c8869ff73bc18bad3732ba359cef1e60ad8f961f9'
Warning: PyVB is not installed!!

file: 025ba9ce4a2118a9ca7b115c8869ff73bc18bad3732ba359cef1e60ad8f961f9
MD5: 01f03f3cc923583a5157243f2a90879d
sha1: 8ccc56a8c89053314bc409948a5f1f040624ed5
sha256: 025ba9ce4a2118a9ca7b115c8869ff73bc18bad3732ba359cef1e60ad8f961f9
size: 41939 bytes
version: 1.4
Binary: False
Linearized: False
Encrypted: False
updates: 1
Objects: 49
Streams: 8
MIA: 11
Comments: 0
Errors: 0

Version 0:
  Catalog: 32
  Info: 1
  Objects (46): [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46]
  Streams (7): [40, 45, 6, 11, 33, 38, 43]
    Included (5): [6, 11, 33, 38, 43]
  Objects with MIA (13): [10, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28]

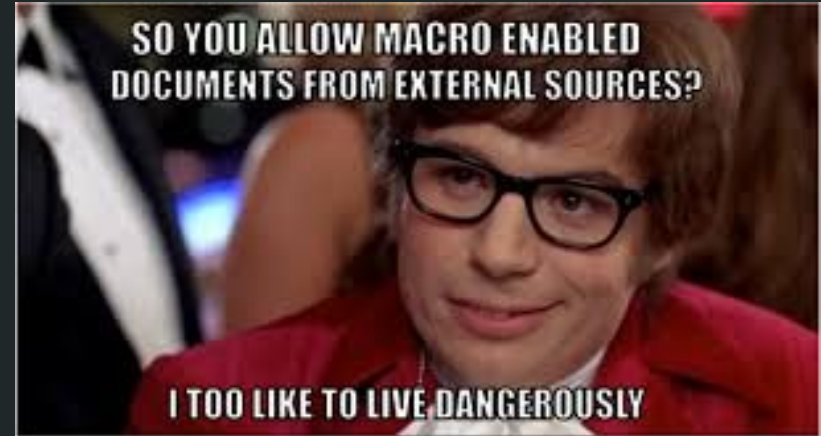
Version 1:
  Catalog: 32
  Info: 1
  Objects (3): [1, 32, 47]
  Streams (1): [47]
    Included (0): []
```


Dump the Raw Content from the Suspicious Object Using pdf-parser

```
pdf-parser.py -o <#> --raw -f <filename with full directory>
```

```
remux@remux:~/Documents/analysis$ pdf-parser.py -o 10,17,18,19,20,21,22,23,24,25,26,27,28 -f /home/remux/Documents/analysis/025ba9ce4a2118a9ca7b115c8869ff73bc16bad3732ba359cef1e60ad8f961f9'  
obj 10 0  
Type: /Annot  
Referencing:  
  
<<  
  /Type /Annot  
  /Subtype /Link  
  /Rect [33.7500000 2132 1649.25000 2355.50000 ]  
  /Border [0 0 0]  
  /A  
    <<  
      /Type /Action  
      /S /URI  
      /URI (https://ttraff.me/wix?keyword=death+in+tehran+parable)  
    >>  
>>  
  
obj 17 0  
Type: /Annot  
Referencing:  
  
<<  
  /Type /Annot  
  /Subtype /Link  
  /Rect [33.7500000 715.250000 1649.25000 727.250000 ]  
  /Border [0 0 0]  
  /A  
    <<  
      /Type /Action  
      /S /URI  
      /URI (http://raseguf.lionstationers.com/uploads/1/3/1/4/131408899/2552184.pdf)  
    >>  
>>  
  
obj 18 0  
Type: /Annot  
Referencing:  
  
<<  
  /Type /Annot  
  /Subtype /Link  
  /Rect [33.7500000 689.750000 1649.25000 701.750000 ]  
  /Border [0 0 0]
```

Detecting and Analyzing Malicious Office Macros Using *oletools*



Detect VBA macros and embedded Flash objects

```
oleid <filename with full directory>
```

```
Filename: 9cafe1ff820182f2d33d662bc3b4018caf27c49d50242573f9620f06001c582f.sample
```

Indicator	Value	Risk	Description
File format	Generic OLE file / Compound File (unknown format)	info	Unrecognized OLE file. Root CLSID: - None
Container format	OLE	info	Container type
Application name	Microsoft Office Word	info	Application name declared in properties
Properties code page	1252: ANSI Latin 1; Western European (Windows)	info	Code page used for properties
Author	HANH-PC	info	Author declared in properties
Encrypted	False	none	The file is not encrypted
VBA Macros	Yes, suspicious	HIGH	This file contains VBA macros. Suspicious keywords were found. Use olevba and mraptor for more info.
XLM Macros	No	none	This file does not contain Excel 4/XLM macros.
External Relationships	0	none	External relationships such as remote templates, remote OLE objects, etc

Analyzing Identified Macros

oledump.py <filename with full directory>

```
remnux@remnux:~/Desktop/userform$ oledump.py userform.doc
1:      113  '\x01CompObj'
2:     4096  '\x05DocumentSummaryInformation'
3:     4096  '\x05SummaryInformation'
4:     4096  '1Table'
5:    23902  'Data'
6:      525  'Macros/PROJECT'
7:      95   'Macros/PROJECTwm'
8: M  10027  'Macros/VBA/ThisDocument'
9:      7279  'Macros/VBA/_VBA_PROJECT'
10: M  15955  'Macros/VBA/cowkeeper'
11:      841  'Macros/VBA/dir'
12: m  1158  'Macros/VBA/discord'
13:      97   'Macros/discord/\x01CompObj'
14:     291  'Macros/discord/\x03VBFrame'
15:      98   'Macros/discord/f'
16:     112  'Macros/discord/i01/\x01CompObj'
17:    7476  'Macros/discord/i01/f'
18:      68   'Macros/discord/i01/o'
19:      0    'Macros/discord/o'
20:   57094  'WordDocument'
```

```
remnux@remnux:~/Desktop/userform$ oledump.py -s 17 userform.doc
00000000: 00 04 2C 00 4A 0C 02 0C FB 8D F8 00 02 00 00 00 ...J.....
00000010: 04 80 00 00 03 00 00 00 02 00 00 00 00 7D 00 00 .....}..
00000020: F8 1F 00 00 9E 02 00 00 00 00 00 00 00 00 00 00 .....
00000030: 01 00 00 00 FC 1C 00 00 00 01 68 6F 00 00 F4 1C .....ho....
00000040: E5 09 00 00 06 00 00 80 02 00 00 00 44 00 00 00 .....D...
00000050: 00 00 1C 00 CC 1C 00 80 70 69 6E 65 61 6C 00 00 .....pineal..
00000060: E3 0A 00 00 22 04 00 00 20 20 20 20 4F 45 4C 6F ..... " ... 0ELO
00000070: 3F 41 75 48 75 51 2E 42 73 44 4D 50 4F 45 67 34 ?AuHuQ.BsDMPOEg4
00000080: 46 41 65 48 27 4D 27 2E 73 67 69 48 75 4C 4B 6D FAeH'M'.sgIHuLkm
00000090: 4F 45 6F 34 46 41 69 48 73 51 65 40 74 3D 66 40 0Eo4FAiHsQe@t=f@
000000A0: 76 49 76 49 76 49 76 49 76 49 76 49 56 51 65 48 vIvIvIvIvIvIVQeH
000000B0: 3E 3F 51 73 3D 3D 3D 3D 4F 45 70 45 55 41 65 48 >?Qs====0EpEUAeH
000000C0: 4D 4E 64 45 65 2C 3D 73 4F 45 70 3D 41 49 4C 49 MNdEe,=s0Ep=AILI
000000D0: 76 49 76 49 76 49 74 41 40 33 55 3F 40 33 55 3E vIvIvItA@3U?@3U>
000000E0: 4D 4F 72 3D 60 4E 70 45 47 34 6D 4D 4E 55 50 3D MOr=`NpEG4mMNUP=
000000F0: 60 3E 46 41 40 33 56 3F 3D 4D 27 2E 4E 3E 41 3E `>FA@3V?=M'.N>A>
00000100: 4F 4C 2B 3F 4D 4F 72 3D 60 4B 69 42 73 44 67 41 0L+?MOr=`KiBsDgA
```

```
remnux@remnux:~/Desktop/userform$ oledump.py -s a -v userform.doc | grep pineal
periapt = discord.playbill.pineal.ControlTipText
```

Checking the file with olevba

```
olevba <filename with full directory> OR olevba --reveal <filename with full directory>
```

```
-----  
VBA MACRO discord.frm  
in file: userform.doc - OLE stream: 'Macros/VBA/discord'  
-----  
(empty macro)  
-----  
VBA FORM Variable "b'playbill'" IN 'userform.doc' - OLE stream: 'Macros/discord'  
-----  
None  
-----  
VBA FORM Variable "b'pineal'" IN 'userform.doc' - OLE stream: 'Macros/discord/i01'  
-----  
b'0'
```

Resources

- <https://www.linkedin.com/pulse/static-malware-analysis-pdf-files-dummies-real-life-demos-raghul-c>
- <https://isc.sans.edu/diary/28894>
- https://opensource.adobe.com/dc-acrobat-sdk-docs/pdfstandards/PDF32000_2008.pdf
- <https://www.thecyberyeti.com/post/identifying-userforms-with-oledump-and-olevba>
- <https://sansorg.egnyte.com/dl/3ydBhha67l>